```
|0001|  @  -------------
|0002|  @ Código de teste
|0003|  @  -------------
|0004|
|0005|  func void fib(int n):
             [0005, 0001] (0001,        FUNC) {func}
             [0005, 0006] (0017,        VOID) {void}
         Var_type = VOID
             [0005, 0011] (0007,          ID) {fib}
             [0005, 0014] (0003,BEGIN_PARAM) {(}
             [0005, 0015] (0012,         INT) {int}
         Var_type = INT
         Var_type_r = Var_type
             [0005, 0019] (0007,          ID) {n}
         Decl_Var = Var_type_r ID
         Parameters = Decl_Var
             [0005, 0020] (0004,   END_PARAM) {)}
             [0005, 0021] (0005,   BEGIN_SCP) {:}
|0006|  int a = 0, int b = 1, int i = 0, int aux
             [0006, 0005] (0012,         INT) {int}
         Var_type = INT
         Var_type_r = Var_type
             [0006, 0009] (0007,          ID) {a}
         Decl_Var = Var_type_r ID
             [0006, 0011] (0009,    OP_ATRIB) {=}
             [0006, 0013] (0031,   CONST_INT) {0}
         Numeric_Const = CONST_INT
         Const = Numeric_Const
         Operand = Const
         Er = Operand
         E = Er
         Decl_Var_r = Decl_Var OP_ATRIB E
             [0006, 0014] (0008,   SEPARATOR) {,}
             [0006, 0016] (0012,         INT) {int}
         Var_type = INT
         Var_type_r = Var_type
             [0006, 0020] (0007,          ID) {b}
         Decl_Var = Var_type_r ID
         Decl_Var_List = Decl_Var_r SEPARATOR Decl_Var
             [0006, 0022] (0009,    OP_ATRIB) {=}
             [0006, 0024] (0031,   CONST_INT) {1}
         Numeric_Const = CONST_INT
         Const = Numeric_Const
         Operand = Const
         Er = Operand
         E = Er
         Decl_Var_r = Decl_Var_List OP_ATRIB E
             [0006, 0025] (0008,   SEPARATOR) {,}
```

```
          [0006, 0027] (0012,        INT) {int}
      Var_type = INT
      Var_type_r = Var_type
          [0006, 0031] (0007,         ID) {i}
      Decl_Var = Var_type_r ID
      Decl_Var_List = Decl_Var_r SEPARATOR Decl_Var
          [0006, 0033] (0009,   OP_ATRIB) {=}
          [0006, 0035] (0031,  CONST_INT) {0}
      Numeric_Const = CONST_INT
      Const = Numeric_Const
      Operand = Const
      Er = Operand
      E = Er
      Decl_Var_r = Decl_Var_List OP_ATRIB E
          [0006, 0036] (0008,  SEPARATOR) {,}
          [0006, 0038] (0012,        INT) {int}
      Var_type = INT
      Var_type_r = Var_type
          [0006, 0042] (0007,         ID) {aux}
|0007|  while (a + b) < n:
      Decl_Var = Var_type_r ID
      Decl_Var_List = Decl_Var_r SEPARATOR Decl_Var
      Decl_Var_r = Decl_Var_List
      Cmd = Decl_Var_r
          [0007, 0005] (0038,  INS_WHILE) {while}
          [0007, 0011] (0003,BEGIN_PARAM) {(}
          [0007, 0012] (0007,         ID) {a}
      Operand = ID
      Er = Operand
      E = Er
          [0007, 0014] (0020,     OP_ADD) {+}
      Opr = OP_ADD
          [0007, 0016] (0007,         ID) {b}
      Operand = ID
      Er = Operand
      E = E Opr Er
          [0007, 0017] (0004,  END_PARAM) {)}
      Er = BEGIN_PARAM E END_PARAM
      E = Er
          [0007, 0019] (0023,   OP_RELAT) {<}
      Opr = OP_RELAT
          [0007, 0021] (0007,         ID) {n}
      Operand = ID
      Er = Operand
      E = E Opr Er
          [0007, 0022] (0005,  BEGIN_SCP) {:}
|0008|  if i > 0:
          [0008, 0009] (0036,     INS_IF) {if}
```

```
        If_r = INS_IF
            [0008, 0012] (0007,          ID) {i}
        Operand = ID
        Er = Operand
        E = Er
            [0008, 0014] (0023,    OP_RELAT) {>}
        Opr = OP_RELAT
            [0008, 0016] (0031,   CONST_INT) {0}
        Numeric_Const = CONST_INT
        Const = Numeric_Const
        Operand = Const
        Er = Operand
        E = E Opr Er
            [0008, 0017] (0005,   BEGIN_SCP) {:}
|0009|  show(",")
            [0009, 0013] (0034,    INS_SHOW) {show}
            [0009, 0017] (0003,BEGIN_PARAM) {(}
            [0009, 0018] (0029,   CONST_STR) {","}
        Param_r = CONST_STR
        Show_Param = Param_r
            [0009, 0021] (0004,   END_PARAM) {)}
|0010|  end
        Show = INS_SHOW BEGIN_PARAM Show_Param END_PARAM
        Cmd = Show
        Cmds = Cmd
            [0010, 0009] (0006,     END_SCP) {end}
|0011|  if i <= 1:
        Cond = If_r E BEGIN_SCP Cmds END_SCP
        Cmd = Cond
            [0011, 0009] (0036,      INS_IF) {if}
        If_r = INS_IF
            [0011, 0012] (0007,          ID) {i}
        Operand = ID
        Er = Operand
        E = Er
            [0011, 0014] (0023,    OP_RELAT) {<=}
        Opr = OP_RELAT
            [0011, 0017] (0031,   CONST_INT) {1}
        Numeric_Const = CONST_INT
        Const = Numeric_Const
        Operand = Const
        Er = Operand
        E = E Opr Er
            [0011, 0018] (0005,   BEGIN_SCP) {:}
|0012|  show(i)
            [0012, 0013] (0034,    INS_SHOW) {show}
            [0012, 0017] (0003,BEGIN_PARAM) {(}
            [0012, 0018] (0007,          ID) {i}
```

```
        Param_r = ID
        Show_Param = Param_r
            [0012, 0019] (0004,   END_PARAM) {)}
|0013|  end
        Show = INS_SHOW BEGIN_PARAM Show_Param END_PARAM
        Cmd = Show
        Cmds = Cmd
            [0013, 0009] (0006,     END_SCP) {end}
|0014|  else:
        Cond = If_r E BEGIN_SCP Cmds END_SCP
        Cmd = Cond
            [0014, 0009] (0035,    INS_ELSE) {else}
            [0014, 0013] (0005,   BEGIN_SCP) {:}
|0015|  aux = a + b
            [0015, 0013] (0007,          ID) {aux}
            [0015, 0017] (0009,    OP_ATRIB) {=}
            [0015, 0019] (0007,          ID) {a}
        Operand = ID
        Er = Operand
        E = Er
            [0015, 0021] (0020,      OP_ADD) {+}
        Opr = OP_ADD
            [0015, 0023] (0007,          ID) {b}
|0016|  show(aux)
        Operand = ID
        Er = Operand
        E = E Opr Er
        Attrib = ID OP_ATRIB E
        Cmd = Attrib
            [0016, 0013] (0034,    INS_SHOW) {show}
            [0016, 0017] (0003,BEGIN_PARAM) {(}
            [0016, 0018] (0007,          ID) {aux}
        Param_r = ID
        Show_Param = Param_r
            [0016, 0021] (0004,   END_PARAM) {)}
|0017|  b = a + b
        Show = INS_SHOW BEGIN_PARAM Show_Param END_PARAM
        Cmd = Show
            [0017, 0013] (0007,          ID) {b}
            [0017, 0015] (0009,    OP_ATRIB) {=}
            [0017, 0017] (0007,          ID) {a}
        Operand = ID
        Er = Operand
        E = Er
            [0017, 0019] (0020,      OP_ADD) {+}
        Opr = OP_ADD
            [0017, 0021] (0007,          ID) {b}
|0018|  a = b - a
```

```
          Operand = ID
          Er = Operand
          E = E Opr Er
          Attrib = ID OP_ATRIB E
          Cmd = Attrib
              [0018, 0013] (0007,        ID) {a}
              [0018, 0015] (0009,   OP_ATRIB) {=}
              [0018, 0017] (0007,        ID) {b}
          Operand = ID
          Er = Operand
          E = Er
              [0018, 0019] (0020,    OP_ADD) {-}
          Opr = OP_ADD
              [0018, 0021] (0007,        ID) {a}
|0019|  end
          Operand = ID
          Er = Operand
          E = E Opr Er
          Attrib = ID OP_ATRIB E
          Cmd = Attrib
          Cmds = Cmd
          Cmds = Cmd Cmds
          Cmds = Cmd Cmds
          Cmds = Cmd Cmds
              [0019, 0009] (0006,    END_SCP) {end}
|0020|  i = 1 + i
          Cond = INS_ELSE BEGIN_SCP Cmds END_SCP
          Cmd = Cond
              [0020, 0005] (0007,        ID) {i}
              [0020, 0007] (0009,   OP_ATRIB) {=}
              [0020, 0009] (0031,  CONST_INT) {1}
          Numeric_Const = CONST_INT
          Const = Numeric_Const
          Operand = Const
          Er = Operand
          E = Er
              [0020, 0011] (0020,    OP_ADD) {+}
          Opr = OP_ADD
              [0020, 0013] (0007,        ID) {i}
|0021|  end
          Operand = ID
          Er = Operand
          E = E Opr Er
          Attrib = ID OP_ATRIB E
          Cmd = Attrib
          Cmds = Cmd
          Cmds = Cmd Cmds
          Cmds = Cmd Cmds
```

```
        Cmds = Cmd Cmds
            [0021, 0005] (0006,     END_SCP) {end}
|0022|  end
        Loop = INS_WHILE E BEGIN_SCP Cmds END_SCP
        Cmd = Loop
        Cmds = Cmd
        Cmds = Cmd Cmds
            [0022, 0001] (0006,     END_SCP) {end}
|0023|
|0024|
|0025|  func void main():
        Decl_Func = FUNC Var_type ID BEGIN_PARAM Parameters END_PARAM
BEGIN_SCP Cmds END_SCP
        List_Func = Decl_Func
            [0025, 0001] (0001,        FUNC) {func}
            [0025, 0006] (0017,        VOID) {void}
        Var_type = VOID
            [0025, 0011] (0002,        MAIN) {main}
            [0025, 0015] (0003,BEGIN_PARAM) {(}
            [0025, 0016] (0004,   END_PARAM) {)}
            [0025, 0017] (0005,   BEGIN_SCP) {:}
|0026|  int n
            [0026, 0005] (0012,         INT) {int}
        Var_type = INT
        Var_type_r = Var_type
            [0026, 0009] (0007,          ID) {n}
|0027|  input(n)
        Decl_Var = Var_type_r ID
        Decl_Var_r = Decl_Var
        Cmd = Decl_Var_r
            [0027, 0005] (0033,   INS_INPUT) {input}
            [0027, 0010] (0003,BEGIN_PARAM) {(}
            [0027, 0011] (0007,          ID) {n}
        Param_r = ID
        Input_Param = Param_r
            [0027, 0012] (0004,   END_PARAM) {)}
|0028|  fib(n)
        Input = INS_INPUT BEGIN_PARAM Input_Param END_PARAM
        Cmd = Input
            [0028, 0005] (0007,          ID) {fib}
            [0028, 0008] (0003,BEGIN_PARAM) {(}
            [0028, 0009] (0007,          ID) {n}
        Operand = ID
        Er = Operand
        E = Er
        Call_Parameters = E
            [0028, 0010] (0004,   END_PARAM) {)}
|0029|  end
```

```
Call_Func = ID BEGIN_PARAM Call_Parameters END_PARAM
Cmd = Call_Func
Cmds = Cmd
Cmds = Cmd Cmds
Cmds = Cmd Cmds
    [0029, 0001] (0006,    END_SCP) {end}
Decl_Main = FUNC Var_type MAIN BEGIN_PARAM END_PARAM BEGIN_SCP Cmds
END_SCP
Sinicial = List_Func Decl_Main

  -------- ACC ---------
```